A Divide and Conquer Direct Differentiation Approach for Multibody System Sensitivity Analysis

This manuscript is without final correction - for corrected manuscript refer:
www.springerlink.com/content/ah025763j4674851

Rudranarayan M. Mukherjee, Kishor D. Bhalerao and Kurt S. Anderson
Department of Mechanical, Aeronautical, and Nuclear Engineering
Rensselaer Polytechnic Institute
Troy NY 12180
email: mukher@rpi.edu bhalek@rpi.edu anderk5@rpi.edu

Abstract

In the design and analysis of multibody dynamics systems, sensitivity analysis is a critical tool for good design decisions. Traditional Direct Differentiation methods can be computationally expensive with complexity as large as $O(n^4 + n^2m^2 + nm^3)$ where n is the the number of generalized coordinates in the system and m is the number of algebraic constraints. In this paper, a Divide and Conquer approach is presented for efficient sensitivity analysis of multibody systems with general topologies. This method requires minimal data storage. This approach uses a binary tree structure to traverse the topology of the system and recursively generate the sensitivity data in $O(n)$ and $O(log(n))$ complexity for serial and parallel implementation. The constraints (in constrained systems) are enforced at the acceleration level. This approach is a good alternative to existing methodologies as it is fairly simple to implement for general topologies and is computationally efficient.

1 Introduction

Multibody design is an iterative process and computationally taxing. Sensitivity analysis is a useful tool which significantly reduces the iterative nature of design by helping make intelligent guesses for the design parameters. However determining sensitivity terms is a non-trivial task given the complexity of governing equations of motions for the simplest of multibody systems. Consequently sensitivity analysis continues to be an important thrust area.

Finite difference approximations for sensitivity analysis though easy straight forward and easy to implement suffers from a number of difficulties. These include the methods sensitivity to parameter perturbation size and system stiffness [1],[3]. Analytical methods such as adjoint variable method [4]-[5], direct differentiation [6]---[9] and automatic differentiation do not suffer from the problems faced by the numerical methods.

With the adjoint variable method, a set of adjoint equations is introduced to represent the variations of the state. The advantage of using these methods is explicit calculation of sensitivity terms is avoided. The number of equations are roughly proportional to the number of objective functions. This is desirable when the number of design variables is large as compared to the objective functions. However the implementation of these methods is complex and a large amount of data has to be stored for the forward problem. The large number of I/O operations slows down the speed significantly [6],[10]. Another source of error is backward temporal integration necessary for calculation of adjoint variables. The adaptive nature of integrators calls for interpolation to obtain all values at matching time steps. Besides this, numerical stability for adjoint variable methods remains an open question [4],[11].

Direct differentiation methods are conceptually easiest to understand. They systematically applies the chain rule of differentiation to obtain analytical expressions for sensitivity terms. The number of integrated equations is roughly equal to the number of state variables plus design variables. The major advantage of these methods is higher numerical stability and relative insensitivity of solution accuracy to parameter perturbations. Implementation approaches for direct differentiation vary with different formulations of equations of motion. Newton-Euler is the most frequently used method [6]-[9]-[12]. Although the formulation of sensitivity equations is straight forward, the resultant is a set of computationally demanding *differential algebraic equations*. Consequently, the cost of computation of sensitivity terms depends upon the algorithm used for solving the equations of motion.

The analytical methods described above are all capable of calculating the sensitivity derivatives. However the costs involved in each method can vary greatly. For example in a system with *g* design variables and *m* independent algebraic constraint equations, the adjoint variable method produces a smaller system of *(n+m+g)* differential algebraic equations, whereas the direct differentiation counterpart involves (n+m)(g+1) differential algebraic equations (DAE). [13] gives a O(n+m) direct differentiation algorithm for a multibody system with *n* generalized coordinates and *m* independent algebraic constraints for sensitivity analysis.

In this paper, a Divide and Conquer direct differentiation approach is presented for efficient sensitivity analysis of multibody systems with general topologies. This method requires minimal data storage as compared to adjoint variable methods that require storing the states of the system at every temporal integration step of the simulation. This approach uses a binary tree structure to traverse the topology of the system and generate the sensitivity data in $O(n)$ and $O(log(n))$ complexity for serial and parallel implementation respectively. The sensitivity data is accurate to integration error making this approach a good alternative to existing methodologies as it is fairly simple to implement for general topologies and is computationally efficient. The methodology presented here is a derived work from the *(i)* divide and conquer algorithm [14] and *(ii)* the orthogonal complement based divide and conquer algorithm [15].

2 Sensitivity Problem Formulations

The objective of sensitivity analysis is to quantify the sensitivity of a desired objective function to the change in certain design or control variable value. This information is useful to identify the robustness of a design as well as tolerances on system performance with respect to variations in design variable values. For multibody dynamics systems the objective function *J* is often an explicit function of design variable(s) *p* as well as state variables *q,u*. The state variables may also be explicit functions of the design variable(s). Further, the dependency of the state variables on the design variable(s) is coupled between all the state variables of the system. Thus, the sensitivity equation of the objective function *J* with respect to design variable *p* can be written as

$$\nabla J = \frac{\partial J}{\partial p} + \sum_{r=1}^{n} \left(\frac{\partial J}{\partial q_r}\frac{dq_r}{dp_j} + \frac{\partial J}{\partial u_r}\frac{du_r}{dp_j} + \frac{\partial J}{\partial \dot{u}_r}\frac{d\dot{u}_r}{dp_j} \right) \tag{1}$$

It is clear from the above equation that the sensitivity analysis requires the generation of the dependencies of the state and state derivatives on the design variable(s). Generating this dependency information can be computationally expensive since the state and state derivative variables are highly coupled for a multibody system. This expense is alleviated somewhat as there exists the following relations

$$\left.\frac{d\boldsymbol{q}_r}{d\boldsymbol{p}_j}\right|_{t=\tau+dt} = \int_{t=\tau}^{t=\tau+dt} \left.\frac{d\dot{\boldsymbol{q}}_r}{d\boldsymbol{p}_j}\right|_{t=\tau} dt + \left.\frac{d\boldsymbol{q}_r}{d\boldsymbol{p}_j}\right|_{t=\tau} \tag{2}$$

$$\left.\frac{d\boldsymbol{u}_r}{d\boldsymbol{p}_j}\right|_{t=\tau+dt} = \int_{t=\tau}^{t=\tau+dt} \left.\frac{d\dot{\boldsymbol{u}}_r}{d\boldsymbol{p}_j}\right|_{t=\tau} dt + \left.\frac{d\boldsymbol{u}_r}{d\boldsymbol{p}_j}\right|_{t=\tau} \tag{3}$$

Thus, the task reduces to that of finding $d\dot{\boldsymbol{u}}_r/dp_j$ at every instant in the simulation and substituting it back into the above relations to generate the other terms. Now, in the state space form, the equations of motion of a general multibody system reduces to

$$\boldsymbol{\mathcal{M}}_{n\times n}\dot{\boldsymbol{u}}_{n\times 1} = \boldsymbol{\mathcal{K}}_{n\times 1} \tag{4}$$

The above present a coupled set of n equations where n is the number of degrees of freedom of the system, $\boldsymbol{\mathcal{M}}$ is the populated mass matrix, $\dot{\boldsymbol{u}}$ is the column matrix of the unknown time derivatives of generalized speeds and $\boldsymbol{\mathcal{K}}$ is the column matrix of the forces on the system including state dependent inertia forces. Using a direct approach, the above equations can be differentiated to generate the desired $\dot{\boldsymbol{u}}$ values as

$$\frac{d\boldsymbol{\mathcal{M}}\dot{\boldsymbol{u}}}{dp_j} = \frac{d\boldsymbol{\mathcal{K}}}{dp_j} \tag{5}$$

$$\Rightarrow [\boldsymbol{\mathcal{M}}_{n\times n}]\frac{d\dot{\boldsymbol{u}}}{dp_j}_{n\times 1} = \frac{\partial\boldsymbol{\mathcal{K}}}{\partial p_j} + \frac{\partial\boldsymbol{\mathcal{K}}}{\partial \boldsymbol{q}}\frac{d\boldsymbol{q}}{dp_j} + \frac{\partial\boldsymbol{\mathcal{K}}}{\partial \boldsymbol{u}}\frac{d\boldsymbol{u}}{dp_j} - [\frac{\partial\boldsymbol{\mathcal{M}}}{\partial p_j} + \frac{\partial\boldsymbol{\mathcal{M}}}{\partial \boldsymbol{q}}\frac{d\boldsymbol{q}}{dp_j} + \frac{\partial\boldsymbol{\mathcal{M}}}{\partial \boldsymbol{u}}\frac{d\boldsymbol{u}}{dp_j}]\dot{\boldsymbol{u}} \tag{6}$$

The direct method incurs large computational expense associated with the differentiations $O(n^2) - O(n^3)$ and the decomposition of the n coupled equations at $O(n^3)$ complexity for direct methods. For even small values of n, this cost can become prohibitive. Unless some efficient method is introduces to reduce the cost, generating sensitivity information for multibody systems can quickly become the bottle neck in the design analysis process.

3 Analytical Preliminaries

Consider an articulated multibody system of rigid bodies connected together by kinematic joints. Consider two representative bodies *Body* k and *Body* $k+1$ of the articulated body system as shown in figure (1). Let the *Body* k be connected to its parent and child bodies through the joints J_1^k and J_2^k where the parent body is *Body* $k-1$ and the child body is *Body* $k+1$. Similarly, the two joints on *Body* $k+1$ are joints J_1^{k+1} and J_2^{k+1}. These joints allow relative motion between bodies through the joint degrees of freedom as well as the transmission of constraint forces through the constrained degrees of freedom. In the absence of a kinematic joint, two bodies can move with respect to each other through 6 degrees of freedom. So the motion map between the bodies is a rank 6 matrix. A kinematic joint constraints the relative motion between two bodies, allowing only certain degrees of freedom (dof) while constraining out the remain. Thus the kinematic joint partitions the 6 dimensional relative motion map between two bodies into the joint motion map H which is of dimension $6 \times dof$ and its orthogonal complement D of dimension $6 \times (6 - dof)$. The joint allows relative motion in the space spanned by the columns of the joint motion map H. The joint cannot support a constraint load in the space spanned by H. However, the constrained degrees of freedom are mapped by the columns of D and the joint can support constraint loads in the space spanned by D. For example, with a spherical joint, the translational degrees of motion are constrained while the rotational degrees of freedom are maintained. Hence the corresponding maps maybe given by

$$H^{k/(k+1)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad D^{k/(k+1)} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{7}$$

From a linear algebra point of view, the joint motion map H can also be interpreted as the $6 \times dof$ matrix that maps the dof generalized speeds u at the joint into a 6×1 column matrix of spatial relative velocity at the joint.

It is apparent from their definitions, the orthogonal complement $D^{k/(k+1)}$ and the joint motion map $H^{k/(k+1)}$ of any representative kinematic joint between two representative bodies k and $k+1$, satisfy the following orthogonality relation

$$H^{k/(k+1)^T} \cdot D^{k/(k+1)} = D^{k/(k+1)^T} \cdot H^{k/(k+1)} = 0 \tag{8}$$

Although for the discussion presented here each body is assumed to have only two joints, the discussion can be easily extended to bodies with an arbitrary number of joints. Each body possess its own mass and inertia properties, with vectors from the body's center of mass to the joint locations which include geometric constants such as lengths. The relative position and orientation of bodies is modelled through the use of

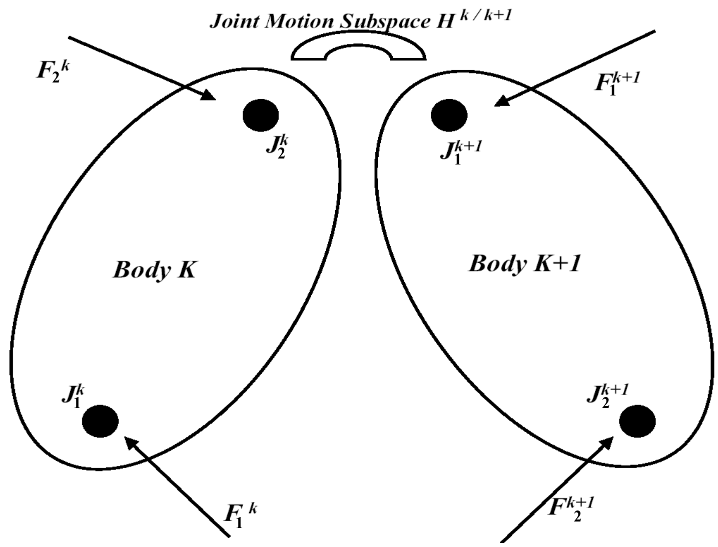

Figure 1: Representative bodies of a multibody system

generalized coordinates q and the kinematics expressed in terms of generalized speeds u and time derivatives of the generalized speeds i.e. $\dot{u}$. The generalized speeds characterized the motion of the systems and are comprised of an invertible linear combination of the time derivatives of the generalized coordinates i.e.

$$u = A\dot{q} + B \quad \text{Where } A^{-1} \text{ exists} \tag{9}$$

In the analytical treatment presented here, direction cosine matrices and transformation between different basis are not shown explicitly. Appropriate basis transformations have to be taken into account for an implementation of this algorithm. Also, this algorithm uses a redundant mixed set of coordinates, viz. Cartesian coordinates and relative coordinates, throughout the derivation. The mixed set of coordinates offer certain advantages within this formulation and have been used in [16]-[17] for rigid body dynamics.

In the most general form, the equations of motion of each body using a spatial Newton-Euler formulation can be expressed as

$$\mathcal{M}_0^k \mathcal{A}_0^k = \mathcal{F}_0^k \tag{10}$$

$$\text{where} \quad \mathcal{M}_0^k = \begin{bmatrix} I_0^k & \underline{0} \\ \underline{0} & m^k \end{bmatrix} \quad \text{while} \quad \mathcal{A}_0^k = \begin{bmatrix} \alpha_0^k \\ a_0^k \end{bmatrix} \quad \text{and} \quad \mathcal{F}_0^k = \begin{bmatrix} \tau_0^k \\ f_0^k \end{bmatrix} \tag{11}$$

In the above, the subscript 0 denotes the center of mass of the body, while superscript k indicates that the quantity is associated with representative body k. The total spatial load acting at the center of mass consists of state dependent active loads such as actuators, loads from potential fields as well as constraint forces arising from the joints. These constraint forces depend on the dynamics of the entire system and hence introduce coupling between the equations of motion of all bodies in the system. The active forces on the other hand are uncoupled and can be calculated independently on each body based on the state of the system. Thus, above equation(8) can be written with $\mathcal{F}_0^k$ expressed explicitly in terms of the known active loads $\mathcal{F}_a^k$ and the unknown constraint loads arising from joints J_1^k and J_2^k as $\mathcal{F}_1^k$ and $\mathcal{F}_2^k$ as

$$\mathcal{M}_0^k \mathcal{A}_0^k = \mathcal{S}^{k_0/k_1} \mathcal{F}_1^k + \mathcal{S}^{k_0/k_2} \mathcal{F}_2^k + \mathcal{F}_a^k \tag{12}$$

$$\text{And} \quad \mathcal{S}^{k_0/k_1} = \begin{bmatrix} \underline{U} & (r^{k_0/k_1})\times \\ \underline{0} & \underline{U} \end{bmatrix} \qquad \mathcal{S}^{k_0/k_2} = \begin{bmatrix} \underline{U} & (r^{k_0/k_2})\times \\ \underline{0} & \underline{U} \end{bmatrix} \tag{13}$$

From the above equations, the expressions for spatial accelerations at the two joints can be expressed as

$$\mathcal{A}_1^k = \zeta_{11}^k \mathcal{F}_1^k + \zeta_{12}^k \mathcal{F}_2^k + \zeta_{13}^k \tag{14}$$

$$\mathcal{A}_2^k = \zeta_{21}^k \mathcal{F}_1^k + \zeta_{22}^k \mathcal{F}_2^k + \zeta_{23}^k \tag{15}$$

In the above, the matrices ζ_{11}^k and ζ_{22}^k are Symmetric Positive Definite (SPD) and the matrices ζ_{12}^k and ζ_{21}^k are symmetric transposes of each other. The matrices ζ_{13}^k and ζ_{23}^k include the terms that are state dependent (for e.g. the active forces) and can be calculated independently on each body. Similar expressions for body $k+1$ can be written as

$$\mathcal{A}_1^{k+1} = \zeta_{11}^{k+1}\mathcal{F}_1^{k+1} + \zeta_{12}^{k+1}\mathcal{F}_2^{k+1} + \zeta_{13}^{k+1} \tag{16}$$
$$\mathcal{A}_2^{k+1} = \zeta_{21}^{k+1}\mathcal{F}_1^{k+1} + \zeta_{22}^{k+1}\mathcal{F}_2^{k+1} + \zeta_{23}^{k+1} \tag{17}$$

Let p_j represent any design variable with respect to which the sensitivity of the dynamic system is to be calculated. For a dynamic system, p_j may be a mass or inertia value, geometric constant such as lengths, radii, or active forces among others. Differentiating equations(14-15) with respect to p_j, the following equations are arrived at.

$$\frac{d\mathcal{A}_1^k}{dp_j} = \frac{d\zeta_{11}^k}{dp_j}\mathcal{F}_1^k + \zeta_{11}^k\frac{d\mathcal{F}_1^k}{dp_j} + \frac{d\zeta_{12}^k}{dp_j}\mathcal{F}_2^k + \zeta_{12}^k\frac{d\mathcal{F}_2^k}{dp_j} + \frac{d\zeta_{13}^k}{dp_j} \tag{18}$$
$$\frac{d\mathcal{A}_2^k}{dp_j} = \frac{d\zeta_{21}^k}{dp_j}\mathcal{F}_1^k + \zeta_{21}^k\frac{d\mathcal{F}_1^k}{dp_j} + \frac{d\zeta_{22}^k}{dp_j}\mathcal{F}_2^k + \zeta_{22}^k\frac{d\mathcal{F}_2^k}{dp_j} + \frac{d\zeta_{23}^k}{dp_j} \tag{19}$$
$$\text{With} \quad \frac{d(\mathcal{M}_0^k)^{-1}}{dp_j} = -(\mathcal{M}_0^k)^{-1}\frac{d\mathcal{M}_0^k}{dp_j}(\mathcal{M}_0^k)^{-1} \tag{20}$$

In the above equations, the terms $d\zeta_{ij}^k/dp_j$ can be easily obtained from $d(\mathcal{M}_0^k)/dp_j$ and $d\mathcal{S}^{k_0/k_i}/dp_j$ locally on each body as there is no coupling in these terms from other bodies in the system. These terms are zero when p_j is a design variable which is not local to the body k. The other terms, viz. $d\mathcal{F}_i^k/dp_j$ and $d\mathcal{A}_i^k/dp_j$ (i=1:2) cannot be calculated locally on each body as these depend on the coupling between different bodies in the system. Further, by solving the equations of motion (14-15) at any instant, the terms $\mathcal{F}_i^k$ (i=1:2) are generated. Thus, having solved the equations of motion at any instant, the above equations(18-19) reduce to the following form with the only unknowns at each body being the terms $d\mathcal{F}_i^k/dp_j$ and $d\mathcal{A}_i^k/dp_j$ (i=1:2).

$$\frac{d\mathcal{A}_1^k}{dp_j} = \Phi_{11}^k\frac{d\mathcal{F}_1^k}{dp_j} + \Phi_{12}^k\frac{d\mathcal{F}_2^k}{dp_j} + \Phi_{13}^k \tag{21}$$
$$\frac{d\mathcal{A}_2^k}{dp_j} = \Phi_{21}^k\frac{d\mathcal{F}_1^k}{dp_j} + \Phi_{22}^k\frac{d\mathcal{F}_2^k}{dp_j} + \Phi_{23}^k \tag{22}$$
$$\text{Where} \quad \Phi_{13}^k = \frac{d\zeta_{11}^k}{dp_j}\mathcal{F}_1^k + \frac{d\zeta_{12}^k}{dp_j}\mathcal{F}_2^k + \frac{d\zeta_{13}^k}{dp_j} \tag{23}$$
$$\text{And} \quad \Phi_{13}^k = \frac{d\zeta_{21}^k}{dp_j}\mathcal{F}_1^k + \frac{d\zeta_{22}^k}{dp_j}\mathcal{F}_2^k + \frac{d\zeta_{23}^k}{dp_j} \tag{24}$$
$$\text{With} \quad \Phi_{ij}^k = \frac{d\zeta_{ij}^k}{dp_j} \quad \text{for i , j = 1:2} \tag{25}$$

Thus, the equations(14-15) and equations(21-22) are in the same analytical form, albeit with different quantities in the equations. Further, the equations(21-22) are obtained in the desired form only if the equations(14-15) have already been solved. The procedure outlined in the next section is as follows: First the constraint forces at each joint in the system are obtained by solving the dynamic equations of motion of the system; These constraint forces are then substituted into equations(18-19) to generate equations(21-22); These are now the sensitivity equations of each body that need to be solved. The corresponding equations for body $k+1$ can be expressed as

$$\frac{d\mathcal{A}_1^{k+1}}{dp_j} = \Phi_{11}^{k+1}\frac{d\mathcal{F}_1^{k+1}}{dp_j} + \Phi_{12}^{k+1}\frac{d\mathcal{F}_2^{k+1}}{dp_j} + \Phi_{13}^{k+1} \tag{26}$$
$$\frac{d\mathcal{A}_2^{k+1}}{dp_j} = \Phi_{21}^{k+1}\frac{d\mathcal{F}_1^{k+1}}{dp_j} + \Phi_{22}^{k+1}\frac{d\mathcal{F}_2^{k+1}}{dp_j} + \Phi_{23}^{k+1} \tag{27}$$

Two Handle Generalized Inertia

The relative acceleration between two joint locations on successive bodies k and $k+1$ can be expressed in terms of the joint motion map $H^{k/(k+1)}$ and the generalized speeds at the joint $u^{k/(k+1)}$ as

$$\mathcal{A}_1^{k+1} - \mathcal{A}_2^{k} \quad = \quad H^{k/k+1}\dot{u}^{k/k+1} + \dot{H}^{k/k+1}u^{k/k+1} \tag{28}$$

Differentiating above equations with respect to parameter p_j

$$\frac{d\mathcal{A}_1^{k+1}}{dp_j} - \frac{d\mathcal{A}_2^{k}}{dp_j} \quad = \quad H^{k/k+1}\frac{d\dot{u}^{k/k+1}}{dp_j} + \underbrace{\frac{dH^{k/k+1}}{dp_j}\dot{u}^{k/k+1} + \frac{d\dot{H}^{k/k+1}u^{k/k+1}}{dp_j}}_{Locally\ Generated} \tag{29}$$

$$\Rightarrow \quad \frac{d\mathcal{A}_1^{k+1}}{dp_j} - \frac{d\mathcal{A}_2^{k}}{dp_j} \quad = \quad H^{k/k+1}\frac{d\dot{u}^{k/k+1}}{dp_j} + \Pi \tag{30}$$

$$\text{Where} \quad \Pi \quad = \quad \frac{dH^{k/k+1}}{dp_j}\dot{u}^{k/k+1} + \frac{d\dot{H}^{k/k+1}u^{k/k+1}}{dp_j} \tag{31}$$

In the above equations, the terms indicated as locally generated depend only on the state sensitivities and the locally generated terms. Hence this can be treated as a known. Further, from Newton's second law

$$\mathcal{F}_2^{k} \quad = \quad -\mathcal{F}_1^{k+1} \quad \Rightarrow \quad \frac{d\mathcal{F}_2^{k}}{dp_j} = -\frac{d\mathcal{F}_1^{k+1}}{dp_j} \tag{32}$$

Substituting the expressions for $d\mathcal{A}_2^{k}/dp_j$ and $d\mathcal{A}_1^{k+1}/dp_j$ from equations(22-26) into equation(29) respectively, and using the relation from equation(28) the following expressions can be arrived at

$$\frac{d\mathcal{A}_1^{k+1}}{dp_j} - \frac{d\mathcal{A}_2^{k}}{dp_j} \quad = \quad \Phi_{11}^{k+1}\frac{d\mathcal{F}_1^{k+1}}{dp_j} + \Phi_{12}^{k+1}\frac{d\mathcal{F}_2^{k+1}}{dp_j} + \Phi_{13}^{k+1} - \Phi_{21}^{k}\frac{d\mathcal{F}_1^{k}}{dp_j} - \Phi_{22}^{k}\frac{d\mathcal{F}_2^{k}}{dp_j} - \Phi_{23}^{k} \tag{33}$$

$$\Rightarrow \; [\Phi_{11}^{k+1} + \Phi_{22}^{k}]\frac{d\mathcal{F}_1^{k+1}}{dp_j} \quad = \quad [\Phi_{21}^{k}\frac{d\mathcal{F}_1^{k}}{dp_j} - \Phi_{12}^{k+1}\frac{d\mathcal{F}_2^{k+1}}{dp_j} + \Phi_{23}^{k} - \Phi_{13}^{k+1} + H^{k/(k+1)}\frac{d\dot{u}}{dp_j} + \Pi] \tag{34}$$

Premultiplying equation (34) by ${D^{k/(k+1)}}^T$ and calling on the orthogonality condition between $D^{k/(k+1)}$ and $H^{k/(k+1)}$

$$\begin{aligned}{D^{k/(k+1)}}^T[\Phi_{11}^{k+1} + \Phi_{22}^{k}]\frac{d\mathcal{F}_1^{k+1}}{dp_j} = {D^{k/(k+1)}}^T[\Phi_{21}^{k}\frac{d\mathcal{F}_1^{k}}{dp_j} + \Phi_{23}^{k} - \Phi_{13}^{k+1} - \Phi_{12}^{k+1}\frac{d\mathcal{F}_2^{k+1}}{dp_j} \\ +\Pi] + \underbrace{{D^{k/(k+1)}}^T H^{k/(k+1)}}_{0}\frac{d\dot{u}}{dp_j}\end{aligned} \tag{35}$$

From the definition of the orthogonal complement of joint motion subspace, the constraint force $\mathcal{F}_1^{k+1}$ can be expressed in terms of the measure numbers of the constraint torques and constraint forces as

$$\mathcal{F}_1^{k+1} = D^{k/(k+1)}\mathbf{F}_1^{k+1} \quad \Rightarrow \quad \frac{d\mathcal{F}_1^{k+1}}{dp_j} = \frac{dD^{k/(k+1)}}{dp_j}\mathbf{F}_1^{k+1} + D^{k/(k+1)}\frac{d\mathbf{F}_1^{k+1}}{dp_j} \tag{36}$$

where the constraint force and constraint moment measure numbers ${\tilde{f}_c}^{{k+1}^+}$ and ${\tilde{\tau}_c}^{{k+1}^+}$, respectively, are represented as

$$\mathbf{F}_1^{k+1} = \begin{bmatrix} {\tilde{\tau}_1}^{k+1} \\ {\tilde{f}_1}^{k+1} \end{bmatrix} \tag{37}$$

Substituting this into above equation(35) an expression for $d\mathcal{F}_1^{k+1}/dp_j$ can be arrived at as below

$$\begin{aligned}{D^{k/(k+1)}}^T[\Phi_{11}^{k+1} + \Phi_{22}^{k}](\frac{dD^{k/(k+1)}}{dp_j}\mathbf{F}^{{k+1}^+} + D^{k/(k+1)}\frac{d\mathbf{F}^{{k+1}^+}}{dp_j}) = \\ {D^{k/(k+1)}}^T[\Phi_{21}^{k}\frac{d\mathcal{F}_1^{k}}{dp_j} + \Phi_{23}^{k} - \Phi_{13}^{k+1} - \Phi_{12}^{k+1}\frac{d\mathcal{F}_2^{k+1}}{dp_j} + \Pi]\end{aligned} \tag{38}$$

$$\Rightarrow \frac{d\mathcal{F}_1^{k+1}}{dp_j} = -\frac{d\mathcal{F}_2^k}{dp_j} = \mathcal{X}\,[\Phi_{21}^k \frac{d\mathcal{F}_1^k}{dp_j} + \Phi_{23}^k - \Phi_{12}^{k+1}\frac{d\mathcal{F}_2^{k+1}}{dp_j} - \Phi_{13}^{k+1}] - \frac{dD^{k/(k+1)}}{dp_j}\mathbf{F}^{k+1^+} \quad (39)$$

$$\text{where} \quad \mathcal{X} = D^{k/k+1}([D^{k/k+1}]^T[\Phi_{22}^k + \Phi_{11}^{k+1}]D^{k/k+1})^{-1}[D^{k/k+1}]^T \quad (40)$$

Substituting this expression for $d\mathcal{F}_1^{k+1}/dp_j$ and $d\mathcal{F}_2^k/dp_j$ into equation(21-27) the following are obtained

$$\frac{d\mathcal{A}_1^k}{dp_j} = \Phi_{11}^{asm}\frac{d\mathcal{F}_1^k}{dp_j} + \Phi_{12}^{asm}\frac{d\mathcal{F}_2^{k+1}}{dp_j} + \Phi_{13}^{asm} \quad (41)$$

$$\frac{d\mathcal{A}_2^{k+1}}{dp_j} = \Phi_{21}^{asm}\frac{d\mathcal{F}_1^k}{dp_j} + \Phi_{22}^{asm}\frac{d\mathcal{F}_2^{k+1}}{dp_j} + \Phi_{23}^{asm} \quad (42)$$

In the above, the super-script asm indicates that bodies k and $k+1$ have been coupled together to form a resulting assembly asm. In substituting the expression for the derivative of the constraint load at the common joint, the equations of the derivatives of the spatial accelerations of the two bodies are coupled together to form the corresponding equations of the resulting assembly. In the resulting assembly, the two joints that connect the assembly to its parent and child bodies (or assemblies) are J^{k^+} and J^{k+1^-}. The Φ_{ij}^{asm} are the inertia coupling terms of the assembly of bodies k and $k+1$ given by

$$\Phi_{11}^{asm} = [\Phi_{11}^k - \Phi_{12}^k\mathcal{X}\Phi_{21}^k] \quad (43)$$

$$\Phi_{12}^{asm} = [\Phi_{12}^k\mathcal{X}\Phi_{12}^{k+1}] \quad (44)$$

$$\Phi_{13}^{asm} = [\Phi_{13}^k - \Phi_{12}^k\mathcal{X}\Phi_{13}^{k+1}] \quad (45)$$

$$\Phi_{21}^{asm} = [\Phi_{21}^{k+1}\mathcal{X}\Phi_{21}^k] \quad (46)$$

$$\Phi_{22}^{asm} = [\Phi_{22}^{k+1} - \Phi_{21}^{k+1}\mathcal{X}\Phi_{12}^{k+1}] \quad (47)$$

$$\Phi_{23}^{asm} = [\Phi_{23}^{k+1} + \Phi_{21}^{k+1}\mathcal{X}\Phi_{23}^{k+1}] \quad (48)$$

Hierarchic Assembly-Disassembly

In the previous section a set of recursive formulae are derived that are used to coupled together the sensitivity equations of two consecutive bodies to form the corresponding equations of the resulting assembly. In the associated manipulations, the two bodies are coupled together to form an assembly by expressing the derivative of the intermediate (common) joint constraint load with respect to the design variable in terms of the corresponding derivatives of the constraint forces at the other two handles. This process can now be repeated for all bodies in the system where the equations of two successive bodies or assemblies are coupled together using the recursive formulae to obtain the corresponding equations of the resulting assembly. This process works hierarchically exploiting the same structure as that of a binary tree.

This process begins at the level of individual bodies of the system. Adjacent bodies of the system are hierarchically assembled to construct a binary tree as shown in Figure (2). Individual bodies that make up the system form the leaf nodes of the binary tree. The sensitivity equations of motion of a pair of bodies are coupled together using the recursive set of formulae (43-48) to form the corresponding equations of the resulting assembly. The resulting assembly now corresponds to a node of the next level in the binary tree. Working along the binary tree in this hierarchic assembly processes, only a single assembly is left at the root node of the binary tree. The root node corresponds to the two-handle representation of the entire articulated system modelled as a single assembly. The sensitivity equations of this root node can be expressed as

$$\frac{d\mathcal{A}_1^{sys}}{dp_j} = \Phi_{11}^{sys}\frac{d\mathcal{F}_1^{sys}}{dp_j} + \Phi_{12}^{sys}\frac{d\mathcal{F}_2^{sys}}{dp_j} + \Phi_{13}^{sys} \quad (49)$$

$$\frac{d\mathcal{A}_2^{sys}}{dp_j} = \Phi_{21}^{sys}\frac{d\mathcal{F}_1^{sys}}{dp_j} + \Phi_{22}^{sys}\frac{d\mathcal{F}_2^{sys}}{dp_j} + \Phi_{23}^{sys} \quad (50)$$

Here the superscript *sys* is used to denote the whole system being represented as a single entity as the root node of the binary tree. In this case, the handles 1 and 2 of this entity are the boundary joints of the articulated system. Similarly the derivatives of the spatial constraint loads are those of the spatial constraint loads arising from the interaction of the system with its boundaries. The above represent two sets of equations in terms of four sets of unknowns i.e the derivatives of the spatial accelerations at the boundary joints $d\mathcal{A}_1^{sys}/dp_j$, $d\mathcal{A}_2^{sys}/dp_j$ and the derivatives of the corresponding constraint loads and $d\mathcal{F}_1^{sys}/dp_j$ and $d\mathcal{F}_2^{sys}/dp_j$. Consider the three following scenarios that may arise for a system.

Free floating

This case corresponds to a system which is free floating, i.e. there are no kinematic joints connecting the system to the inertial frame. In the absence of any kinematic joints at either boundary, there are no constraint forces that can act on the system at the boundaries. In this case, in the equations(49-50) the constraint loads terms are all zero and hence their derivatives are always zero. From this, the derivatives of the spatial accelerations can be easily solved as

$$\frac{d\mathcal{A}_1^{sys}}{dp_j} = \Phi_{13}^{sys} \quad \text{and} \quad \frac{d\mathcal{A}_2^{sys}}{dp_j} = \Phi_{23}^{sys} \tag{51}$$

Anchored at one end by kinematic joint

In this case, the system is connected to the inertial frame by a kinematic joint at one end while the other end is free floating. For such a system, there is no constraint load acting at the free end and in the equations(49-50) the term $d\mathcal{F}_2^{sys}/dp_j = 0$ and hence its derivative is also always zero. However at the other end, the system will experience a constraint load because of the presence of the kinematic joint and its derivative needs to be accounted for. The equations in this case reduce to

$$\begin{aligned} \frac{d\mathcal{A}_1^{sys}}{dp_j} &= \Phi_{11}^{sys}\frac{d\mathcal{F}_1^{sys}}{dp_j} + \Phi_{13}^{sys} && (52)\\ \frac{d\mathcal{A}_2^{sys}}{dp_j} &= \Phi_{21}^{sys}\frac{d\mathcal{F}_1^{sys}}{dp_j} + \Phi_{23}^{sys} && (53) \end{aligned}$$

From the definition of the kinematic joint and its joint motion map, there exist the following kinematic relations:

$$\begin{aligned} \frac{d\mathcal{A}_1^{sys}}{dp_j} &= H^1\frac{d\dot{u}^1}{dp_j} + \underbrace{\frac{dH^1}{dp_j}\dot{u}^1 + \frac{d(\dot{H}^1u^1)}{dp_j}}_{Locally\ Generated} && (54)\\ \Rightarrow \quad \frac{d\mathcal{A}_1^{sys}}{dp_j} &= H^1\frac{d\dot{u}^1}{dp_j} + \Pi^{sys} && (55)\\ \text{Where} \quad \Pi^{sys} &= \frac{dH^1}{dp_j}\dot{u}^1 + \frac{d(\dot{H}^1u^1)}{dp_j} && (56) \end{aligned}$$

Further, from the definition of the orthogonal complement of the joint motion map, the constraint load at the handle can be expressed as

$$\mathcal{F}_1^{sys} = D^1\mathbf{F}_1^{sys} \Rightarrow \quad \frac{d\mathcal{F}_1^{sys}}{dp_j} = \frac{dD^1}{dp_j}\mathbf{F}_1^{sys} + D^1\frac{d\mathbf{F}_1^{sys}}{dp_j} \tag{57}$$

Substituting the above equations(54-57) into equations (52-53) the following is arrived at.

$$\begin{aligned} \frac{d\mathcal{A}_1^{sys}}{dp_j} &= H^1\frac{d\dot{u}^1}{dp_j} + \Pi^{sys} = \Phi_{11}^{sys}[\frac{dD^1}{dp_j}\mathbf{F}_1^{sys} + D^1\frac{d\mathbf{F}_1^{sys}}{dp_j}] + \Phi_{13}^{sys} && (58)\\ \Rightarrow \quad H^1\frac{d\dot{u}^1}{dp_j} &= \Phi_{11}^{sys}[\frac{dD^1}{dp_j}\mathbf{F}_1^{sys} + D^1\frac{d\mathbf{F}_1^{sys}}{dp_j}] + \Phi_{13}^{sys} - \Pi^{sys} && (59) \end{aligned}$$

Using the orthogonality relation between H^1 and D^1, the derivative of the generalized speed at the joint as well as that of the constraint load can be solved from equation(59) as

$$\underbrace{{D^1}^T H^1}_{0} \frac{d\dot{u}^1}{dp_j} = {D^1}^T \Phi_{11}^{sys} D^1 \frac{d\mathbf{F}_1^{sys}}{dp_j} + {D^1}^T [\Phi_{11}^{sys} \frac{dD^1}{dp_j} \mathbf{F}_1^{sys} + \Phi_{13}^{sys} - \Pi^{sys}] \tag{60}$$

$$\Rightarrow \frac{d\mathcal{F}_1^{sys}}{dp_j} = -D^1[(D^1)^T \Phi_{11}^{sys} D^1]^{-1} (D^1)^T [\Phi_{11}^{sys} \frac{dD^1}{dp_j} \mathbf{F}_1^{sys} + \Phi_{13}^{sys} - \Pi^{sys}] \tag{61}$$

$$\Rightarrow \frac{du^1}{dp_j} = H^1[(H^1)^T (\Phi_{11}^{sys})^{-1} H^1]^{-1} (H^1)^T [\Phi_{11}^{sys} \frac{dD^1}{dp_j} \mathbf{F}_1^{sys} + \Phi_{13}^{sys} - \Pi^{sys}] \tag{62}$$

Substituting the above equations(61-62) into equations(52-53), the derivatives of the boundary spatial accelerations $d\mathcal{A}_1^k/dp_j$ and $d\mathcal{A}_1^k/dp_j$ can be easily calculated.

Anchored at both ends by kinematic joints

In this case, the system is connected to the inertial frame by a kinematic joint at both ends and the system reduces to a kinematically closed loop topology. For such a system, there are constraint load acting at both the ends due to the kinematic joints. In this case the sensitivity equations for the system remains

$$\frac{d\mathcal{A}_1^{asm}}{dp_j} = \Phi_{11}^{asm} \frac{d\mathcal{F}_1^{asm}}{dp_j} + \Phi_{12}^{asm} \frac{d\mathcal{F}_2^{asm}}{dp_j} + \Phi_{13}^{k} \tag{63}$$

$$\frac{d\mathcal{A}_2^{asm}}{dp_j} = \Phi_{21}^{asm} \frac{d\mathcal{F}_1^{asm}}{dp_j} + \Phi_{22}^{asm} \frac{d\mathcal{F}_2^{asm}}{dp_j} + \Phi_{23}^{asm} \tag{64}$$

Similar to the previous situation, the following kinematic relations exist between the boundary joints and their joint motion maps.

$$\frac{d\mathcal{A}_1^{sys}}{dp_j} = H^1 \frac{d\dot{u}^1}{dp_j} + \underbrace{\frac{dH^1}{dp_j}\dot{u}^1 + \frac{d(\dot{H}^1 u^1)}{dp_j}}_{Locally\ Generated} \quad \text{and} \quad \frac{d\mathcal{A}_2^{sys}}{dp_j} = H^2 \frac{d\dot{u}^2}{dp_j} + \underbrace{\frac{dH^2}{dp_j}\dot{u}^2 + \frac{d(\dot{H}^2 u^2)}{dp_j}}_{Locally\ Generated} \tag{65}$$

$$\Rightarrow \frac{d\mathcal{A}_1^{sys}}{dp_j} = H^1 \frac{d\dot{u}^1}{dp_j} + \Pi_1^{sys} \quad \text{and} \quad \frac{d\mathcal{A}_2^{sys}}{dp_j} = H^2 \frac{d\dot{u}^2}{dp_j} + \Pi_2^{sys} \tag{66}$$

$$\text{Where} \quad \Pi_1^{sys} = \frac{dH^1}{dp_j}\dot{u}^1 + \frac{d(\dot{H}^1 u^1)}{dp_j} \quad \text{and} \quad \Pi_2^{sys} = \frac{dH^2}{dp_j}\dot{u}^2 + \frac{d(\dot{H}^2 u^2)}{dp_j} \tag{67}$$

Further, from the definition of the orthogonal complement of the joint motion map, the constraint load at the handle can be expressed as

$$\mathcal{F}_1^{sys} = D^1 \mathbf{F}_1^{sys} \quad \Rightarrow \quad \frac{d\mathcal{F}_1^{sys}}{dp_j} = \frac{dD^1}{dp_j} \mathbf{F}_1^{sys} + D^1 \frac{d\mathbf{F}_1^{sys}}{dp_j} \tag{68}$$

$$\mathcal{F}_2^{sys} = D^2 \mathbf{F}_2^{sys} \quad \Rightarrow \quad \frac{d\mathcal{F}_2^{sys}}{dp_j} = \frac{dD^2}{dp_j} \mathbf{F}_2^{sys} + D^2 \frac{d\mathbf{F}_2^{sys}}{dp_j} \tag{69}$$

Substituting the equations (66) into equations (63-64) and absorbing the terms Π_i^{sys} into the Φ_{i3}^{sys} term $(i = 1:2)$, one obtains,

$$H^1 \frac{d\dot{u}^1}{dp_j} = \Phi_{11}^{asm} \frac{d\mathcal{F}_1^{asm}}{dp_j} + \Phi_{12}^{asm} \frac{d\mathcal{F}_2^{asm}}{dp_j} + \Phi_{13}^{asm} \tag{70}$$

$$H^2 \frac{d\dot{u}^2}{dp_j} = \Phi_{21}^{asm} \frac{d\mathcal{F}_1^{asm}}{dp_j} + \Phi_{22}^{asm} \frac{d\mathcal{F}_2^{asm}}{dp_j} + \Phi_{23}^{asm} \tag{71}$$

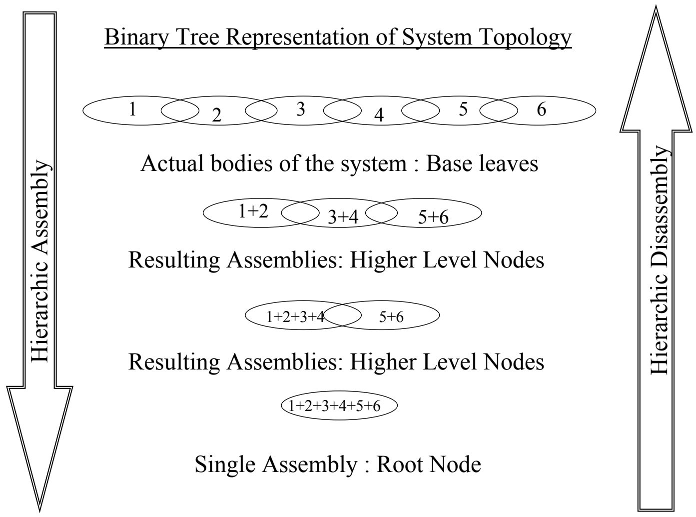

Figure 2: The Hierarchic Assembly and Disassembly Process using Binary Tree Structure

Multiplying the above equations by $(D^1)^T$ and $(D^2)^T$ respectively, and calling on the orthogonality relation, the following is obtained.

$$\overbrace{(D^1)^T H^1}^{0} \frac{du^{1}}{dp_j} = (D^1)^T[\Phi_{11}^{asm}\frac{d\mathcal{F}_1^{asm}}{dp_j} + \Phi_{12}^{asm}\frac{d\mathcal{F}_2^{asm}}{dp_j} + \Phi_{13}^{asm}] = 0 \quad (72)$$

$$\underbrace{(D^2)^T H^2}_{0} \frac{du^2}{dp_j} = (D^2)^T[\Phi_{21}^{asm}\frac{d\mathcal{F}_1^{asm}}{dp_j} + \Phi_{22}^{asm}\frac{d\mathcal{F}_2^{asm}}{dp_j} + \Phi_{23}^{asm}] = 0 \quad (73)$$

Substituting the expressions for the derivatives of the constraint loads from equation(68-69) into the equations (72-73) one obtains

$$(D^1)^T\Phi_{11}^{sys}D^1\frac{d\mathbf{F}_1^{sys}}{dp_j} + (D^1)^T\Phi_{12}^{sys}D^2\frac{d\mathbf{F}_2^{sys}}{dp_j} + (D^2)^T[\Phi_{11}^{sys}\frac{dD^1}{dp_j}\mathbf{F}_1^{sys} + \Phi_{12}^{sys}\frac{dD^2}{dp_j}\mathbf{F}_2^{sys} + \Phi_{13}^{sys}] = 0 \quad (74)$$

$$(D^2)^T\Phi_{21}^{sys}D^1\frac{d\mathbf{F}_1^{sys}}{dp_j} + (D^2)^T\Phi_{22}^{sys}D^2\frac{d\mathbf{F}_2^{sys}}{dp_j} + (D^2)^T[\Phi_{21}^{sys}\frac{dD^1}{dp_j}\mathbf{F}_1^{sys} + \Phi_{22}^{sys}\frac{dD^2}{dp_j}\mathbf{F}_2^{sys} + \Phi_{23}^{sys}] = 0 \quad (75)$$

In these equations, the terms $(D^1)^T\Phi_{11}^{sys}D^1$ and $(D^2)^T\Phi_{22}^{sys}D^2$ are symmetric positive definite (SPD) matrices and there is no problem associated with their inversion. For notational convenience, the above equations can be represented compactly in matrix form as

$$\begin{bmatrix}\chi_{11} & \chi_{12}\\ \chi_{21} & \chi_{22}\end{bmatrix}\begin{bmatrix} d\mathbf{F}_1^{sys}/dp_j \\ d\mathbf{F}_2^{sys}/dp_j\end{bmatrix} = -\begin{bmatrix}\chi_{13}\\ \chi_{23}\end{bmatrix} \quad (76)$$

where the corresponding χ_{ij} can be derived from above equation. The matrix in (76) is also SPD with $\chi_{12} = \chi_{21}^T$. Having solved the above equations for the values of $d\mathbf{F}_1^{sys}/dp_j$ and $d\mathbf{F}_2^{sys}/dp_j$, the corresponding expression for $d\mathcal{F}_1^{sys}/dp_j$ and $d\mathcal{F}_2^{sys}/dp_j$ can be obtained from equation(68-69). At this point, the derivatives of both constraint loads on the boundary joints are known. Consequently, the equation(63-64) of the root node can be solved to obtain the derivatives of the spatial accelerations $d\mathcal{A}_1^{sys}/dp_j$ and $d\mathcal{A}_1^{sys}/dp_j$ at the corresponding joints.

Thus in all three cases, the derivatives of the spatial accelerations and the constraint loads at the boundary joints can be calculated. This initiates the hierarchic disassembly process. The derivatives of the spatial accelerations and the constraint loads generated by solving the sensitivity equations of an assembly are identically the values of the derivatives of the spatial accelerations and the constraint loads on one handle on each of the two constituent assemblies. From these known quantities, the sensitivity equations of the constituent assemblies can be solved to obtain the derivatives of the spatial accelerations and that of the constraint loads at the connecting joint. For example, for a representative assembly made from *Body k* and *Body k+1*, the sensitivity equations are given by equations (41-42). On solving these equations the quantities $d\mathcal{A}_1^k/dp_j$, $d\mathcal{A}_2^{k+1}/dp_j$, $d\mathcal{F}_1^k/dp_j$ and $d\mathcal{F}_2^{k+1}/dp_j$ are generated. These quantities are then substituted into the sensitivity equations of the constituent sub-assemblies say for *Body k* and *Body k+1*. Thus knowing the values of $d\mathcal{A}_1^k/dp_j$, $d\mathcal{F}_1^k/dp_j$, equations (21-22) can be solved, while from $d\mathcal{A}_2^{k+1}/dp_j$ and $d\mathcal{F}_2^{k+1}/dp_j$ equations (26-27) can also be solved. This process is repeated in a hierarchic disassembly of the binary tree where the known derivatives of the boundary conditions are used to solve the sensitivity equations of the immediate subassemblies, until derivatives of the spatial acceleration and constraint forces on all bodies in the system are calculated.

The local analytical derivatives used in the algorithm are temporally invariant and exact. These are generated once during a preprocessing step and introduced into the algorithm as an input. This algorithm works in six sweeps of the system, traversing the system topology like a binary tree. The first four sweeps are associated with formulating and solving the equations of motion for the forward dynamics problem. The next two sweeps are associated with the sensitivity analysis and correspond to the hierarchic assembly and the hierarchic disassembly processes respectively. These last two sweeps may additionally be performed concurrently with the final two sweeps of the forward dynamics formulation, if the concurrent formulation is preferred, where the sensitivity sweeps lags behind the equations of motion by one level of the binary tree.

4 Conclusions

In this paper a new efficient method is presented for sensitivity analysis of multibody systems. The method uses a direct differentiation approach and implements it in a divide and conquer scheme. The method maps the topology of the system to a binary tree and generates the sensitivity information using several traverses of this binary tree. The computational complexity of this algorithm is expected to be $O(n)$ and $O(log(n))$ in serial and parallel implementation, respectively. The method works in tandem with the forward dynamics problem. Consequently there is no excessive data storage and no backward integration in this scheme. Further the limited amount of associated differentiation is carried out analytically and exactly. Thus the method does not suffer from numerical issues associated with perturbations in design variables. The method is robust and does not suffer from numerical dependency issues associated with singular configurations. Implementation of sample test cases to validate the method and an extended implementation for generalized topologies are current focus of research of the authors.

Acknowledgement

This work was funded by the NSF NIRT Grant Number 0303902. The authors would like to thank the funding agency for their support.

Lock Step Method

Two handle equations of motion can be written as :

$$\mathcal{M}_0^k \mathcal{A}_0^k = \mathcal{S}^{k_0/k_1}\mathcal{F}_1^k + \mathcal{S}^{k_0/k_2}\mathcal{F}_2^k + \mathcal{F}_a^k \tag{77}$$

$$\text{And} \quad \mathcal{S}^{k_0/k_1} = \begin{bmatrix} \underline{U} & (r^{k_0/k_1})\times \\ \underline{0} & \underline{U} \end{bmatrix} \qquad \mathcal{S}^{k_0/k_2} = \begin{bmatrix} \underline{U} & (r^{k_0/k_2})\times \\ \underline{0} & \underline{U} \end{bmatrix} \tag{78}$$

From the above equations, the expressions for spatial accelerations at the two joints can be expressed as

$$\mathcal{A}_1^k = \zeta_{11}^k \mathcal{F}_1^k + \zeta_{12}^k \mathcal{F}_2^k + \zeta_{13}^k \tag{79}$$

$$\mathcal{A}_2^k = \zeta_{21}^k \mathcal{F}_1^k + \zeta_{22}^k \mathcal{F}_2^k + \zeta_{23}^k \tag{80}$$

In the above, the matrices ζ_{11}^k and ζ_{22}^k are Symmetric Positive Definite (SPD) and the matrices ζ_{12}^k and ζ_{21}^k are symmetric transposes of each other. The matrices ζ_{13}^k and ζ_{23}^k include the terms that are state dependent (for e.g. the active forces) and can be calculated independently on each body. Similar expressions for body $k+1$ can be written as

$$\mathcal{A}_1^{k+1} = \zeta_{11}^{k+1} \mathcal{F}_1^{k+1} + \zeta_{12}^{k+1} \mathcal{F}_2^{k+1} + \zeta_{13}^{k+1} \tag{81}$$

$$\mathcal{A}_2^{k+1} = \zeta_{21}^{k+1} \mathcal{F}_1^{k+1} + \zeta_{22}^{k+1} \mathcal{F}_2^{k+1} + \zeta_{23}^{k+1} \tag{82}$$

Let p_j represent any design variable with respect to which the sensitivity of the dynamic system is to be calculated. For a dynamic system, p_j may be a mass or inertia value, geometric constant such as lengths, radii, or active forces among others. Differentiating equations(79-80) with respect to p_j, the following equations are arrived at.

$$\frac{d\mathcal{A}_1^k}{dp_j} = \frac{d\zeta_{11}^k}{dp_j}\mathcal{F}_1^k + \zeta_{11}^k\frac{d\mathcal{F}_1^k}{dp_j} + \frac{d\zeta_{12}^k}{dp_j}\mathcal{F}_2^k + \zeta_{12}^k\frac{d\mathcal{F}_2^k}{dp_j} + \frac{d\zeta_{13}^k}{dp_j} \tag{83}$$

$$\frac{d\mathcal{A}_2^k}{dp_j} = \frac{d\zeta_{21}^k}{dp_j}\mathcal{F}_1^k + \zeta_{21}^k\frac{d\mathcal{F}_1^k}{dp_j} + \frac{d\zeta_{22}^k}{dp_j}\mathcal{F}_2^k + \zeta_{22}^k\frac{d\mathcal{F}_2^k}{dp_j} + \frac{d\zeta_{23}^k}{dp_j} \tag{84}$$

$$\text{With} \quad \frac{d(\mathcal{M}_0^k)^{-1}}{dp_j} = -(\mathcal{M}_0^k)^{-1}\frac{d\mathcal{M}_0^k}{dp_j}(\mathcal{M}_0^k)^{-1} \tag{85}$$

Similarly for body $k+1$ the corresponding equations are

$$\frac{d\mathcal{A}_1^{k+1}}{dp_j} = \frac{d\zeta_{11}^{k+1}}{dp_j}\mathcal{F}_1^{k+1} + \zeta_{11}^{k+1}\frac{d\mathcal{F}_1^{k+1}}{dp_j} + \frac{d\zeta_{12}^{k+1}}{dp_j}\mathcal{F}_2^{k+1} + \zeta_{12}^{k+1}\frac{d\mathcal{F}_2^{k+1}}{dp_j} + \frac{d\zeta_{13}^{k+1}}{dp_j} \tag{86}$$

$$\frac{d\mathcal{A}_2^{k+1}}{dp_j} = \frac{d\zeta_{21}^{k+1}}{dp_j}\mathcal{F}_1^{k+1} + \zeta_{21}^{k+1}\frac{d\mathcal{F}_1^{k+1}}{dp_j} + \frac{d\zeta_{22}^{k+1}}{dp_j}\mathcal{F}_2^{k+1} + \zeta_{22}^{k+1}\frac{d\mathcal{F}_2^{k+1}}{dp_j} + \frac{d\zeta_{23}^{k+1}}{dp_j} \tag{87}$$

$$\tag{88}$$

Subtract equation 80 from 81 and the follow the common method of DCA as in my previous papers to generate an expression for the intermediate joint constraint force as

$$F_1^{k+1} = \widehat{W}{\zeta_{21}}^k F_1^k - \widehat{W}{\zeta_{12}}^{k+1} F_2^{k+1} + \widehat{Y} \tag{89}$$

$$\text{where} \quad \widehat{\mathrm{W}} = D^{k/(k+1)}\widehat{X}^{-1}{D^{k/(k+1)}}^T \tag{90}$$

$$\text{and} \quad \widehat{\mathrm{Y}} = \widehat{W}[{\zeta_{23}}^k - {\zeta_{13}}^{k+1} + \dot{H}^{k/(k+1)}u] \tag{91}$$

Substitute the above equation into equations 80 and 81 and then initiate the same process for the hierarchic assembly of sensitivity equations as outlined in the paper. In this case the resulting equations would look like as follows with the constraint force at the boundary joints explicitly present as opposed to being treated as known quantities.

$$\frac{dA_1^k}{dp_j} = \Psi_{11}^{k:k+1}F_1^k + \zeta_{11}^{k:k+1}\frac{dF_1^k}{dp_j} + \Psi_{12}^{k:k+1}F_2^{k+1} + \zeta_{12}^{k:k+1}\frac{dF_2^{k+1}}{dp_j} + \frac{d\Psi_{13}^{k:k+1}}{dp_j} \tag{92}$$

$$\frac{dA_2^{k+1}}{dp_j} = \Psi_{21}^{k:k+1}F_1^k + \zeta_{21}^{k:k+1}\frac{dF_1^k}{dp_j} + \Psi_{22}^{k:k+1}F_2^{k+1} + \zeta_{22}^{k:k+1}\frac{dF_2^{k+1}}{dp_j} + \frac{d\Psi_{23}^{k:k+1}}{dp_j} \tag{93}$$

$$\tag{94}$$

In the above the terms Ψ now have new definitions and contain the inertia coupling terms originating from the expression of the intermediate joint constraint force. These are the new sensitivity equations and not the ones we had used earlier in the paper. These equations allow us the calculate the sensitivities as well as the equations of motion concurrently. The assembly process is apparent from the above and the disassembly is now going to follow in tandem by first solving the equations of motion, generating the constraint forces and substituting the same in the new equations above. This way there are no 6 sweeps but 4 and the sensitivities are generated in tandem with the equations of motion. The only question that arises is whether the matrices are still SPD? My understanding is that it depends primarily on the design variable and in most cases it should still be SPD unless we choose some weird design parameter. Note that these set of equations allow us the work concurrently and is not really a lock step process in terms of traversal of the binary tree. These equations allow us to solve for the equations of motion and sensitivity equations in the order mentioned for an assembly and then move simultaneously to the next level in the binary tree. Thus although the sensitivity solution follows the solution of the equations of motion (and can thus be thought of as lock step), the process does not require the two solutions to lag by one or more levels of the binary tree. I think its best if Kishor can implement this quickly for the double pendulum problem and verify. Should not take more than a couple of days.

References

[1] D. Bestle and P. Eberhard. Analysing and optimizing multibody systems. *Structures and Machines*, 20:67–92, 1992.

[2] K. S. Anderson and Y. H. Hsu. Low operational order analytic sensitivity analysis for tree-type multibody dynamic systems. *Journal of Guidance, Control and Dynamics*, 24(6):1133–1143, Nov.–Dec. 2001.

[3] C. H. Bischof. On the automatic differentiation of computer programs and an application to multibody systems. In *Proceedings of the IUTAM Symposium on Optimization of Mechanical Systems*, pages 41–48, 1996.

[4] D. Bestle and J. Seybold. Sensativity analysiz of constrained optimization in dynamic systems. *Archive of Applied Mechanics*, 62:181–190, 1992.

[5] P. Eberhard. Adjoint variable method for sensitivity analysis of multibody systems interpreted as a continuous, hybrid form of automatic differentiation. In *Proceedings of the 2nd International Workshop on Computational Differentiation*, pages 319–328, Santa Fe, NM, USA, Feb 12-14 1996.

[6] C. O. Chang and P. E. Nikravesh. Optimal design of mechanical systems with constaint violation stabilization method. *Journal of Mechanisms, Transmissions, and Automation in Design*, 107:493–498, Dec. 1985.

[7] T. Tak. *A Recursive Approach to Design Sensitivity Analysis of Multibody Systems Using Direct Differentiation*. PhD thesis, University of Iowa, Iowa City, 1990.

[8] A. Jain and G. Rodrigues. Sensitivity analysis of multibody systems using spatial operators. In *Proceedings of the International Conference on Method and Models in Automation and Robotics (MMAR 2000)*, Miedzyzdroje, Poland, Aug. 28-31 2000.

[9] R. Serban and E. J. Haug. Kinematic and kinetics derivatives for multibody system analyses. *Mechanics of Structures and Machines*, 26(2):145–173, 1998.

[10] J.M. Pagalday, I. Aranburu, A. Avello, and J.G. De Jalon. Multibody dynamics optimization by direct differentiation methods using object oriented programming. In *Proceedings of the IUTAM Symposium on Optimization of Mechanical Systems*, pages 213–220, Stuttgart, Germany, March 26-31 1995.

[11] L.F.P. Etman. *Optimization of multibody systems using approximation concepts*. PhD thesis, Technische Universiteit Eindhoven, The Netherlands., 1997.

[12] J.M.P. Dias and M.S. Pereira. Sensitivity analysis of rigidflexible multibody systems. *Multibody System Dynamics*, 1(3):303–322, 1997.

[13] K. S. Anderson and Y. H. Hsu. 'Order-(n+m)' direct differentiation determination of design sensitivity for constrained multibody dynamic systems. *Structural and Multidisciplinary Optimization*, 26(3–4):171–182, Feb. 2004.

[14] R. Featherstone. A divide-and-conquer articulated body algorithm for parallel $O(\log(n))$ calculation of rigid body dynamics. Part 1: Basic algorithm. *International Journal of Robotics Research*, 18(9):867–875, Sep. 1999.

[15] R. Mukherjee and K. S. Anderson. An orthogonal complement based divide-and-conquer algorithm for constrained multibody systems. *Nonlinear Dynamics*, 2005. Accepted for Publication.

[16] Kim S. S. and VanderPloeg M. J. Generalized and efficient method for dynamic analysis of mechanical systems using velocity transforms. *Journal of Mechanism, Transmissions, and Automation Design*, 108(2):176–182, 1986.

[17] Nikravesh P. E. Systematic reduction of multibody equations to a minimal set. *International Journal of Non-Linear Mechanics*, 25(2-3):143–151, 1990.